Data and Applications Security: Developments and Directions
B. Thuraisingham, R. van de Riet, K. R. Dittrich, and Z. Tari, eds., Kluwer, 2001.

Chapter 29

Trust Management in Distributed Databases

James B. Michael and Leonard T. Gaines
Naval Postgraduate School, Computer Science Department, Monterey, CA

Key words: Trust, Trust Management, Internet Security, Distributed Databases

Abstract: Businesses and the military must be able to incorporate information from a number of sources in different formats to remain competitive. As the need for information increases, more applications are utilizing distributed databases. Data is collected from multiple sources in multiple formats and is combined into data warehouses or datamarts. For example, military applications are incorporating distributed databases to combine sensor information for use in command and control. Intelligent agents can already search the web for information sources. However, issues of interconnectivity among the agents and information sources, data overflow, data validity, and security remain to be addressed. This article addresses the security and data validity issues. Specifically, the article addresses trust management and its application to obtaining information utilizing an inherently untrustworthy medium.

1. INTRODUCTION

The Internet has created an opportunity for organizations to gather more quantitative and qualitative information for decision makers. The ability to analyze information faster and more efficiently than the competition permits organizations to better position themselves in the marketplace so as to react quickly to changes in the business environment.

As applications such as online analytical processing (OLAP) tools become more sophisticated, the need to gather and filter information will become crucial. Soon these tools will begin to incorporate intelligent agents to gather information. These agents can search a distributed system for information, or they can monitor sites, reporting on significant or changing

information. Once the agents obtain data, they can pass it to a data warehouse that can be accessed by the application tools.

The use of intelligent agents and distributed databases raises a number of concerns about trust. The Internet and other distributed systems that encompass two or more administrative domains for security (i.e., enclaves) are inherently untrustworthy. Authentication of users and nodes (e.g., web sites) can be difficult, and the paths that data packets traverse are not always owned or controlled by entities that use them. The presence of viruses, Trojan horses, and hackers also adds to the public's mistrust of distributed systems. How does user know that the information retrieved by a system is from a reputable source? How can a system verify the legitimacy of a node? Can a user trust the owners or users of a particular node in a distributed system?

Concerns associated with trust in distributed databases can be addressed, to some extent, by utilizing a trust-management system. Members of an organization tend not to want to use data and information from sources that they do not trust. The motivation for the work reported here is to explore the extent to which trust-management systems can assist the members of an organization, to decide, based on consideration of policy about trust, whether to access data or information from a particular source in a distributed database system.

2. TRUST AND DISTRIBUTED SYSTEMS

Many believe that cryptography is the key to security on the Internet, but it does not address all of the pertinent security issues. When connecting with a server and exchanging information utilizing secure socket layer (SSL), how do you know that you have connected to the correct server? Site spoofing involves using URLs that are similar to popular web pages in the hopes that people will incorrectly type a URL and land on the rogue site. A good example is Whitehouse.com, which is a pornography site instead of the government site located at Whitehouse.gov. The site may look exactly like the site you want, but unless you open the certificate and compare the name on the certificate to the site, SSL will allow you to transact business with the rogue site.

Intelligent agents can check certificates to validate sites, but how can they determine the accuracy of the information located at the sites? Additionally, how does the agent verify whether a reputable organization issued the certificate? The Internet Information Server can create its own certificate. When downloading information from a web site, how does the agent know whether the information contains malicious code? Additionally, if a client downloads Java applets or Active X code, how does the client know whether

the mobile code is malicious until it is too late to prevent the malicious code from executing?

In summary, the user must form an opinion concerning the extent to which he or she trusts the developers of the downloadable program and the web site that is distributing the program. However, a user must be able to analyze the risks and be knowledgeable enough to make an informed decision on matters of trust, which can be difficult when dealing with complex technical issues. Trust-management systems are designed to assist the user by evaluating the action to be taken, gathering the information required to form a decision, and determining whether the action to be taken is consistent with a policy about trust.

3. TRUST MANAGEMENT

In order for intelligent agents to interact with the Internet, in a secure manner, a methodology must be developed for identifying and validating web sites and their information. One of the methods to accomplish this is to add labels to the web sites that contain their certificates and outline the information contained in each site. Additional labels can attest to a form of validation similar to the Trusted Computer Security Evaluation Criteria (TCSEC) model. These validations can consist of a level of security, organization of data, an evaluation of the sources of information, and possibly insurance information covering the site. Utilizing these labels, organizations would be better able to evaluate the information they are receiving from the Internet. However, a trust-management system would still need to be implemented to ensure that the information gathered from a distributed database system met with certain organization-wide and user-defined trust criteria.

Trust models have been used to mimic human trust, dissect trust into element parts, categorize trust, and assign metrics to trust. The designers of the trust models try to communicate a notion of trust from one entity to another. Since trust is a subjective belief, one must assign a metric to beliefs that will have value when evaluating trust.

According to Gaines, trust management has a number of definitions. (Gaines, L., 2000) Some believe it is the process of translating a trust model into a practical application by combining trust variables associated with authentication with those of integrity and confidentiality. Others believe it is a system for protecting open, decentralized systems by analyzing, codifying, and managing trust decisions.

The authors of the REFEREE trust-management system argue that trust management provides a systematic means for deciding whether a requested action, supported by credentials, conforms to a specific policy. (Chu, Y., Feigenbaum, J., LaMacchia, B., Resnick, P., and Strauss, M., 1997) Another view of trust management is that it is a new philosophy for codifying, ana-

lyzing, and managing decisions about trust, with regarding to the overarching question "Is someone trusted to take action on some object?" (Khare, R. and Rifkin, A., June, 1998, p. 2)

In order to implement a trust-management system with OLAP tools, we must first develop a way to identify all of the principals (i.e., the entities involved). The use of digital certificates within a public-key infrastructure (PKI) is an example of one way to accomplish this. The second step is to list the various elements of the system, and for instance, use external metadata labels. These labels can be bound by a URL to a specific web-based resource. These labels can be in a Platform for Internet Content Selection (PICS) format. The final step is to specify the authorization decisions according to some policy. The REFEREE trust-management system (discussed later) addresses these steps. In addition, REFEREE makes trust decisions based upon a target, a principal, a proposed action, and policy. (Khare, R. and Rifkin, A., June, 1998)

Trust management systems such as REFEREE take as input a subject, action, and statements about the subject, matching these to a module containing the corresponding policy. For each action, there are specific policies that govern which statements are valid.

Khare and Rifkin discuss three types of approaches to framing policies. The first approach based on principal-centric policies, which forward a notion that only certain people can be trusted. The policy-enforcement mechanism checks the clearance of each principal to determine whether that principal can perform an action on an object. Another approach is based on object-centric policy. Handles, tokens, combinations, and cryptographic keys are the essence of object-centric policy. A principal must have a trusted object that represents permission to execute actions on another object. The third approach relies on action-centric policy, that is, policy that specifies that only certain actions can be trusted: the policy-enforcement mechanism must ensure that any action taken by a principal on an object is approved. (Khare, R. and Rifkin, A., 30 November, 1997.)

REFEREE has four major components: the metadata format, the trust protocol, the trust-policy languages, and the execution environment. The REFEREE system was designed to incorporate these four components. (Chu, Y., June 1997) PICS labels contain metadata about the site. The metadata can be queried. The information contained in the metadata is applied to heuristics and trust protocols to determine whether an action is permitted by policy.

The trust-policy languages must be capable of interpreting the various forms of metadata and applying the information to internal trust protocols. The trust protocols consist of gathering all of the necessary information or assertions to determine if a given request complies with a trust policy. The trust protocols process the query on the metadata.

The execution environment is the place where a request is evaluated against a trust policy and the pertinent metadata information. It accepts requests and interprets the truth policies that pertain to the requests. It also triggers the trust protocols to gather the necessary information to make a decision. Then it provides an answer to the request along with an explanation.

For a given user request, REFEREE invokes the appropriate user policy and interpreter module and returns to the host application an answer of whether or not the request complies with the policy. The basic computing unit is a module. It is an executable block of code that processes the input arguments, compares the input to policies, and outputs an answer. The module consists of a policy and zero or more interpreters. Modules can delegate tasks to other modules if necessary. Modules can also be easily added or deleted; they are contained in a module database that cross-references the requested action with the appropriate module and interpreter.

REFEREE is a good trust management system in that it is one of the first to combine all of the categories of trust management into one system. The other system, Microsoft's Authenticode, also combines all of the categories into one system, but its application is limited. Authenticode does not have the flexibility that is inherent in REFEREE. (Chu, Y., 13 June 1997)

4. JØSANG'S TRUST MODEL

An important part of REFEREE is authentication through the use of certificates. However, cryptography does not address issues of trust associated with the public-key infrastructure (PKI).

Jøsang's trust model was developed for use in the authentication of public keys. In an open environment such as the Internet, certificates alone cannot validate authenticity. The trust in the binding of a certificate key and its owner is essential in providing a level of legal culpability (i.e., digital certificates and non-repudiation). The certification authority that created the certificate must also be assessed for trustworthiness. Do they properly check identification before issuing a certificate? The authenticity of a key can be validated with its corresponding public or private key. However, the certificate that holds the key is what needs to be validated.

Jøsang defined trust as a subjective measure: the belief that a system will resist malicious attacks. Trust in humans was defined as the belief that he or she will cooperate and not defect. (Jøsang, A., 1999) In his model, he assumes that the outcome of a transaction depends on whether an agent defects or cooperates. Thus, probabilities are not assigned to possible outcomes. Instead, trust measures are used as input to a decision mechanism.

In Jøsang's trust model the truth-value of a statement must be crisp (i.e., they are either true or false). Whenever the truth of a statement is assessed, it is always done by an individual, and therefore represents a subjec-

tive determination of trust. The belief in a statement cannot be purely binary. Humans do not have perfect knowledge, so it is impossible to know with certainty whether a statement is true or false. We can only have "opinions" about the veracity of a statement. These opinions represent degrees of belief, disbelief, and uncertainty. Jøsang expresses "opinions" mathematically as $b + d + u = 1$ and $b,d,u \in [0,1]$, where b, d, and u represent belief, disbelief, and uncertainty, respectively.

Jøsang's trust model is founded on subjective logic. Subjective logic defines the various logical operators for combining opinions. These operators are conjunction, disjunction, negation, recommendation, and consensus: they are the same operators as those found in classical and subjective logics, but they are applied to trust.

A conjunction of two opinions combines an individual's opinions on two distinct binary statements into one opinion that reflects the belief in the truth of both statements. If x and y are two distinct statements, the conjunction of the belief in x, represented by $W_x = (b_x, d_x, u_x)$ and y represented by $W_y = (b_y, d_y, u_y)$ represents an individual's opinion about both x and y being true.

If we represent the conjunction of an individual's opinions on statements x and y as $W_{x\wedge y}$, then $W_{x\wedge y} = (b_{x\wedge y}, d_{x\wedge y}, u_{x\wedge y})$. In order to compute the conjunction, the individual values of belief, disbelief, and uncertainty must be combined for the opinions on both statements. In contrast, the disjunction operation represents an individual's opinion about statements x or y or both being true, that is, $W_{x\vee y} = (b_{x\vee y}, d_{x\vee y}, u_{x\vee y})$.

A negation of an opinion represents the belief that a statement is false. If W_x represents an opinion, $W_{\neg x}$ represents the negation of W_x such that $W_{\neg x} = (b_{\neg x}, d_{\neg x}, u_{\neg x})$.

Subjective logic can also be used to convey values for recommendation. Recall that trust in humans is the belief that the human will cooperate and not defect. Agent A has an opinion about B's willingness to cooperate and not defect. Agent B has an opinion about a statement or proposition x. A recommendation consists of combining B's opinion about x with A's opinion about B's cooperation, so A can form an opinion about statement x.

An assumption underlying the recommendation operator is that the agents do not defect or change their recommendations depending on whom they interact with. In addition, there is an assumption that the opinions that are recommended are independent. If a chain of recommenders is needed to gain information about a proposition x, it is assumed that only first-hand knowledge is transmitted. If second-hand knowledge is passed as a recommendation, opinion independence is violated. Additionally, the order in which the opinions are combined is significant.

Subjective logic has consensus operators. A consensus operator allows two independent agents to form a consensus opinion based on each agent's individual opinions concerning a proposition x.

Jøsang provides an example of how subjective logic can be used to measure the trust in a certificate. In some PKI architectures, a certificate authority issues certificates containing an individual's public key. If agent *A* knows certification authority *B*'s public key k_b and *B* knows agent *C*'s public key k_c, then *B* can send *C*'s public key to *A* signed by *B*'s private key k_{-1b}. Agent *A* will verify the certificate with *B*'s public key, and if correct, will know that it has received a correct copy of *C*'s public key.

Unfortunately, this exchange does not convey *A*'s trust that it has received a correct copy of *C*'s public key. In order to trust in a certificate, *A* must have an opinion about the validity of *B*'s public key. *A* must also form an opinion on agent cooperation, which measures *A*'s trust in *B* to properly certify other keys. *A* must also evaluate the recommendation of *B* as to the validity of *C*'s public key.

In order to validate the authenticity of the certificate, *A* must first evaluate the recommendation from certification authority *B*. *A* will combine its opinion of *B*'s key authentication with its opinion about *B*'s agent cooperation. This will determine *A*'s opinion about *B*'s capability as a recommender. Then *A* must combine its opinion about *B*'s recommendation ability with *B*'s recommendation about *C*'s public key. (Jøsang, A., 1998)

Jøsang has demonstrated the versatility of his model by showing that it is capable of chaining trust and certificate relationships using multiple recommendation operators. The model also supports measuring trust along multiple trust-paths and combining them into a single representation, and assigning utility values (i.e., weights) to levels of trust.

5. PRACTICAL APPLICATION

In order to provide trust-based decision-making support for applications that rely on distributed databases a combination of Jøsang's public-key-authentication trust model and REFEREE can be used. Such a combination can permit an application to validate a web site and utilize metadata to determine whether the data can be trusted. This section contains a practical application utilizing an OLAP tool. The discussion here is based on a portion of the thesis research conducted by Gaines. (Gaines, L., 2000)

In response to a request generated by the front-end, the OLAP server queries the data source. If additional information is needed, intelligent agents are deployed to collect the pertinent data. When an agent arrives at a web site, it examines the site's metadata; contained in this metadata is the site's certificate. In order to authenticate this site, the certificate is passed to the OLAP server.

The OLAP server, when receiving a certificate, can utilize Jøsang's model to compute a level of trust in the certificate. If a chain of trust is needed to validate the certificate, then the system can generate the queries necessary to collect recommender information. The OLAP server can com-

pute a probability-expectation value and compare this value to a value in user-defined policy. If the certificate is trusted, then additional metadata will need to be collected. Otherwise, the agent will not access that site.

In this scenario, metadata includes a rating level from an outside entity that evaluates the way data is organized, evaluates data sources, and judges an organization's reputation. The agent passes the metadata to the referee system along with a statement such as "can this agent access this web site?" The trust protocol can collect the necessary metadata and pass it to the execution environment. The trust-policy language can then be used to select the syntax to apply so that a policy can be compared to the metadata. The execution environment analyzes the statement, the metadata information, and compares both to a corresponding preset policy about trust. The execution environment returns an answer: access permitted or denied.

Suppose that two different agents pose the same query to different web sites. The query results turn out to be different, even partially inconsistent with one another. The agents each have their own opinions as to the trustworthiness of the sources. However, by combining their opinions using the consensus operator in Jøsang's model, it may be possible for the agents to reduce their combined level of uncertainty about the trustworthiness of the sources.

REFEREE is designed to determine whether an agent should perform a potentially dangerous task, such as downloading unknown Java applets. The agent asks the system for permission to execute a particular task. The system evaluates the task and the metadata information and compares it to a policy about trust. If the REFEREE system trusts the site or the software being downloaded, then it will allow the agent to perform some action on that site or use the software that was downloaded.

The foregoing example is somewhat oversimplified. For example, we ignored the complexities associated with composing heterogeneous trust-management systems. In addition to the need for semantic interoperability between heterogeneous database systems, Hansen, for instance, points out the necessity for both technical and functional interoperability between the public-key infrastructures that are used by the U.S. Department of Defense and other branches of government to manage trust. (Hansen, A., 1999)

6. CONCLUSION

Users of distributed database systems can rely to some extent on trust-management systems, in conjunction with their portfolio of other types of security services, to partially automate both reasoning about and enforcing policy about trust. Instead of placing universal trust in an object or node within a distributed database system, the decision-maker can take steps to gauge the trustworthiness of the object or node, in addition to passing his or her trust in the object or node to another party.

Trust-management systems provide applications with the ability to make informed decisions about actions performed by their distributed databases, including the actions of intelligent agents. Trust-management systems are not a silver bullet for addressing all of the challenges associated with trust-based decision-making in distributed database systems, but they do provide an avenue for managing trust, and hence, managing risk associated with trusting a source of data and information.

Disclaimer

The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distributed reprints for Government purposes not withstanding any copyright annotations thereon.

List of References

Chu, Y., "Trust Management for the World Wide Web," Master's thesis, Massachusetts Institute of Technology, 1997.

Chu, Y., Feigenbaum, J., LaMacchia, B., Resnick, P., and Strauss, M., "REFEREE: Trust Management for Web Applications," [http://www.research.att.com/~bal/papers/www6-referee/www6-referee.html], 1997.

Dousette, P., Danesh, A., and Jones, M., "Command and Control using World Wide Web Technology," [http://turing.acm.org:8005/pubs/citations/proceedings/ada/289524/p212-dousette], 1998.

Gaines, L. T., "Trust and its Ramifications for the DOD Public Key Infrastructure," Master's Thesis, Naval Postgraduate School, 2000.

Hansen, A. P., "Public Key Infrastructure Interoperability: A Security Services Approach to Support Transfer of Trust," Master's thesis, Naval Postgraduate School, 1999.

Jøsang, A., "An Algebra for Assessing Trust in Certification Chains," in *Proceedings of the Network and Distributed Systems Security (NDSS'99) Symposium*, Internet Society, 1999.

Jøsang, A., "A Subjective Metric of Authentication," in *Proceedings of the Fifth European Symposium on Research in Computer Securit)*, Springer-Verlag, 1998.

Jøsang, A., "Trust-Based Decision Making for Electronic Transactions," in *Proceedings of the Fourth Nordic Workshop on Secure Computer Systems*, 1999.

Khare, R., and Rifkin, A., "Trust Management on the World Wide Web," [http://www.firstmonday.dk/issue3_6/khare/], June 1998.

Khare, R. and Rifkin, A., "Weaving a Web of Trust," [http://www.cs.caltech.edu/~adam/local/trust.html], 30 November 1997.